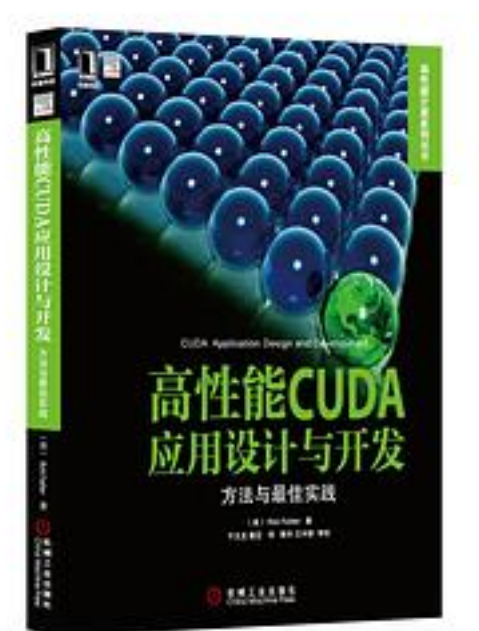


# 高性能CUDA应用设计与开发



[高性能CUDA应用设计与开发 下载链接1](#)

著者:Rob Farber

出版者:机械工业出版社华章公司

出版时间:2013-1-1

装帧:平装

isbn:9787111404460

本书是广受推崇的系统学习高性能CUDA应用开发与设计的经典著作，是美国国家安全实验室资深高性能编程专家多年工作经验结晶，橡树岭国家实验室资深专家鼎力推荐！本书不仅从硬件角度深入解读了CUDA的设计理念和GPGPU硬件的体系结构，而且从软件角度系统讲解了CUDA应用设计与开发的思想、方法、技巧、准则、注意事项和最佳实践。

第1章首先介绍了CUDA的核心概念和编程思想，以及构建与调试CUDA应用所需的工具和方法，然后讲解了有效提高程序性能的CPU编程准则；第2章讲解了CUDA在机器学习与优化中的核心概念与应用，并给出了完整的通用框架；第3章介绍了CUDA的性能分

析工具套件以及性能分析的方法，同时讨论了PCA和NLPCA两种数据挖掘方法；第4章讲解了CUDA的执行模型，深刻揭示了GPU的工作方式和原理；第5章介绍了CUDA提供的多种GPU内存，以及各种内存的优缺点；第6章讲解了高效利用内存的技术；第7章介绍了GPU提供的多种并行方式及其应用；第8章首先讨论了多种CUDA后端设备，以及CUDA如何与Python、Java、R等高级语言交互；第9章讲解了CUDA与图形渲染混合编程；第10章讲解了在云计算和集群环境中使用CUDA的方法和技术细节；第11章介绍了CUDA在高维数据处理、力导向图、交互式工作流、量子化学等现实问题中的应用；第12章为学习CUDA设计了一个综合性的针对实时视频流的应用案例。

## 作者介绍:

Rob Farber，资深高性能编程专家，Irish高端计算中心和美国国家实验室等权威机构的高性能编程技术顾问，同时为多家《财富》世界500强企业提供咨询服务，经验十分丰富，在该领域颇具权威和影响力。他还是一位技术作家，任职于Santa Fe学院，在《Dr. Dobbs' s Journal》《Scientific Computing》等媒体上发表了多篇关于高性能编程的经典技术文章，深受读者喜爱。此外，他还是《财富》美国100强中两家公司的合伙创始人。

## 目录: 译者序

### 序言

### 前言

## 第1章 CUDA入门与编程思想1

### 1.1 源代码与维基1

### 1.2 一个用以区别CUDA与传统程序开发的示例2

### 1.3 选择合适的CUDA API5

### 1.4 CUDA的一些基本概念7

### 1.5 理解首个Runtime Kernel10

### 1.6 GPGPU编程的三条法则11

#### 1.6.1 法则1：将数据放入并始终存储于GPU12

#### 1.6.2 法则2：交给GPGPU足够多的任务12

#### 1.6.3 法则3：注重GPGPU上的数据重用，以避免带宽限制12

### 1.7 大O记号的思想与数据传输13

### 1.8 CUDA和Amdahl定律15

### 1.9 数据并行与任务并行15

### 1.10 混合执行：同时使用CPU和GPU资源16

### 1.11 回归测试与正确性18

### 1.12 静默错误19

### 1.13 调试简介20

### 1.14 UNIX调试方法21

#### 1.14.1 NVIDIA cuda-gdb调试器21

#### 1.14.2 CUDA内存检查器23

#### 1.14.3 通过UNIX ddd界面使用cuda-gdb24

### 1.15 使用Parallel Nsight进行Windows调试25

### 1.16 本章小结27

## 第2章 CUDA在机器学习与优化中的应用28

### 2.1 建模与模拟28

#### 2.1.1 拟合参数化模型29

#### 2.1.2 Nelder-Mead方法30

#### 2.1.3 Levenberg-Marquardt方法30

#### 2.1.4 算法加速31

### 2.2 机器学习与神经网络32

2.3 异或逻辑：一个重要的非线性机器学习问题	33
2.3.1 目标函数示例	35
2.3.2 针对多GPU设备、多CPU处理器的完整仿函数	35
2.3.3 完整Nelder-Mead优化代码的简要讨论	37
2.4 异或逻辑的性能结果	45
2.5 性能讨论	45
2.6 本章小结	48
2.7 C++ NELDER-MEAD代码模板	48
第3章 CUDA工具套件：对PCA、NLPCA进行性能分析	53
3.1 PCA和NLPCA	53
3.1.1 自编码网络	55
3.1.2 用于PCA分析的仿函数示例	56
3.1.3 用于NLPCA分析的示例仿函数	58
3.2 获得基础性能分析数据	60
3.3 gprof：通用UNIX性能分析器	61
3.4 NVIDIA可视化性能分析器：computeprof	62
3.5 Microsoft Visual Studio中的Parallel Nsight	65
3.5.1 Nsight时间表分析	66
3.5.2 NVTX跟踪支持库	67
3.5.3 CUDA API的可扩展性表现	68
3.6 性能调节与分析实用工具（TAU）	70
3.7 本章小结	70
第4章 CUDA执行模型	72
4.1 GPU架构综述	72
4.1.1 线程调度：通过执行配置统筹性能与并行度	74
4.1.2 computeprof中Warp相关值	77
4.1.3 Warp分歧	77
4.1.4 关于Warp分歧的若干准则	78
4.1.5 computeprof中Warp分歧相关值	79
4.2 Warp调度与TLP	79
4.3 ILP：高性能低占用率	80
4.3.1 ILP隐藏算术计算延迟	81
4.3.2 ILP隐藏数据延迟	84
4.3.3 ILP的未来	84
4.3.4 computeprof中指令速率相关值	85
4.4 Little法则	86
4.5 检测限制因素的CUDA工具	87
4.5.1 nvcc编译器	88
4.5.2 启动约束	90
4.5.3 反汇编器	90
4.5.4 PTX Kernel函数	92
4.5.5 GPU模拟器	92
4.6 本章小结	93
第5章 CUDA存储器	94
5.1 CUDA存储器层次结构	94
5.2 GPU存储器	95
5.3 L2缓存	98
5.4 L1缓存	99
5.5 CUDA内存类型	100
5.5.1 寄存器	101
5.5.2 局域内存	101
5.5.3 和局域内存相关的computeprof性能分析参数	102
5.5.4 共享内存	102
5.5.5 和共享内存相关的computeprof性能分析参数	105

5.5.6 常量内存	105
5.5.7 纹理内存	106
5.5.8 和纹理内存相关的computeprof性能分析参数	108
5.6 全局内存	109
5.6.1 常见的整合内存示例	110
5.6.2 全局内存的申请	111
5.6.3 全局内存设计中的限制因素	113
5.6.4 和全局内存相关的computeprof性能分析参数	114
5.7 本章小结	115
第6章 高效使用CUDA存储器	116
6.1 归约	116
6.1.1 归约模板	117
6.1.2 functionReduce.h的测试程序	122
6.1.3 测试结果	126
6.2 使用非规则数据结构	127
6.3 稀疏矩阵和CUSP支持库	131
6.4 图论算法	132
6.5 SoA、AoS以及其他数据结构	134
6.6 分片和分块	135
6.7 本章小结	136
第7章 提高并行度的技巧	137
7.1 CUDA上下文环境对并行度的扩展	137
7.2 流与上下文环境	138
7.2.1 多GPU的使用	139
7.2.2 显式同步	139
7.2.3 隐式同步	141
7.2.4 统一虚拟地址空间	141
7.2.5 一个简单的示例	142
7.2.6 分析结果	144
7.3 使用多个流乱序执行	144
7.3.1 在同一GPU内并发执行Kernel函数的建议	147
7.3.2 隐式并行Kernel的原子操作	147
7.4 将数据捆绑计算	149
7.4.1 手动分割数据	150
7.4.2 映射内存	150
7.4.3 映射内存的工作机制	152
7.5 本章小结	153
第8章 CUDA在所有GPU与CPU程序中的应用	154
8.1 从CUDA到多种硬件后端的途径	155
8.1.1 PGI CUDA x86编译器	155
8.1.2 PGI CUDA x86编译器	157
8.1.3 将x86处理器核心用作流多处理器	159
8.1.4 NVIDIA NVCC编译器	160
8.1.5 Ocelot	160
8.1.6 Swan	161
8.1.7 MCUDA	162
8.2 从其他语言访问CUDA	162
8.2.1 SWIG	162
8.2.2 Copperhead	163
8.2.3 EXCEL	164
8.2.4 MATLAB	164
8.3 支持库	164
8.3.1 CUBLAS	164
8.3.2 CUFFT	165

- 8.3.3 MAGMA174
- 8.3.4 phiGEMM支持库175
- 8.3.5 CURAND176
- 8.4 本章小结177
- 第9章 CUDA与图形渲染混合编程178
- 9.1 OpenGL178
  - 9.1.1 GLUT179
  - 9.1.2 通过OpenGL映射GPU内存179
  - 9.1.3 使用基元重启提升3D处理性能181
- 9.2 框架内各文件的介绍183
  - 9.2.1 Kernel与Perlin Kernel演示的示例代码184
  - 9.2.2 simpleGLmain.cpp文件192
  - 9.2.3 simpleVBO.cpp文件196
  - 9.2.4 callbacksVBO.cpp文件199
- 9.3 本章小结204
- 第10章 在云计算和集群环境中使用CUDA205
- 10.1 消息传递接口205
  - 10.1.1 MPI编程模型206
  - 10.1.2 MPI通信器206
  - 10.1.3 MPI进程号206
  - 10.1.4 主从模式208
  - 10.1.5 点对点模式基础208
- 10.2 MPI通信机制209
- 10.3 带宽211
- 10.4 平衡率212
- 10.5 运行大型MPI程序需要考虑的因素214
  - 10.5.1 初始数据加载的可扩展性214
  - 10.5.2 使用MPI进行计算215
  - 10.5.3 可扩展性检查216
- 10.6 云计算217
- 10.7 代码示例218
  - 10.7.1 数据的产生218
  - 10.7.2 主体代码部分220
- 10.8 本章小结225
- 第11章 CUDA在现实问题中的应用227
- 11.1 高维数据的处理228
  - 11.1.1 PCA/NLPCA228
  - 11.1.2 多维尺度分析229
  - 11.1.3 K均值聚类算法229
  - 11.1.4 期望最大化229
  - 11.1.5 支持向量机230
  - 11.1.6 Bayesian网络230
  - 11.1.7 互信息231
- 11.2 力导向图232
- 11.3 Monte Carlo方法232
- 11.4 分子建模233
- 11.5 量子化学234
- 11.6 交互式工作流234
- 11.7 其他众多的项目235
- 11.8 本章小结235
- 第12章 针对现场实况视频流的应用程序236
- 12.1 机器视觉话题236
  - 12.1.1 3D效果237
  - 12.1.2 肤色区域分割238

12.1.3 边缘检测238  
12.2 FFmpeg239  
12.3 TCP服务器241  
12.4 实况视频流应用程序244  
12.4.1 kernelWave(): 动画Kernel函数244  
12.4.2 kernelFlat(): 在平面渲染图像245  
12.4.3 kernelSkin(): 仅保留肤色区域245  
12.4.4 kernelSobel(): Sobel边缘检测过滤器246  
12.4.5 launch\_kernel()方法247  
12.5 simpleVBO.cpp文件248  
12.6 callbacksVBO.cpp文件248  
12.7 生成与执行代码251  
12.8 展望251  
12.8.1 机器学习252  
12.8.2 Connectome252  
12.9 本章小结253  
12.10 simpleVBO.cpp文件253  
参考文献258  
术语表265  
• • • • • ([收起](#))

[高性能CUDA应用设计与开发\\_下载链接1](#)

## 标签

CUDA

并行

并行计算

programming

计算机

程序设计

C++

计算机科学

## 评论

讲了一些概念性的东西，例子也并未多作解释，想要上手CUDA编程还是得去看NVIDIA文档。但有概念不明白了，可以再来看看这个

-----  
好是好，但是感觉太浮光掠影了些

-----  
嗯。看了两页。。。

-----  
适合有基础的同学做代码优化时使用

-----  
[高性能CUDA应用设计与开发\\_下载链接1](#)

## 书评

这本书不适合初学者，因为内容有一定深度，适合有一定基础的CUDA开发者进行代码优化阶段的提高工具。初学者还是推荐使用《GPU高性能编程 CUDA实战》那本书，那本书上手快，对于深层问题做了较好的省略。等学完那本薄册子再来读这个，效果就会很好了。

-----  
推荐有一定基础的同学阅读本书。书里面设计了各种cuda的应用，如机器学习；而且设计到多GPU， MPI+GPU 还有OpenGL+GPU等比较前沿的应用领域。因此，该书适合已了解cuda及并行计算之后，去进行知识扩展。同时，由于该书设计内容广泛，每一章讲述也相对比较泛，而且有些...

-----  
比较偏工程一些，但是太宽泛，没有深入下去  
比较偏工程一些，但是太宽泛，没有深入下去

比较偏工程一些，但是太宽泛，没有深入下去  
比较偏工程一些，但是太宽泛，没有深入下去  
比较偏工程一些，但是太宽泛，没有深入下去  
比较偏工程一些，但是太宽泛，没有深入下去

-----  
[高性能CUDA应用设计与开发\\_下载链接1](#)